

The example NDDL implementation provided in the two *.nddl files [here](#) is lightly commented and should be self-explanatory.

The Shopping example requires an agent to start at home, buy milk, bananas, and a drill, and finally to finish back at home.

Run the example:

```
% cd $EUROPA_HOME/examples/Shopping
% ant
```

The following figure shows the resulting EUROPA plan (the table is produced by the code in [Shopping.bsh](#)):

Shopping timeline							
Key	Name	ES	start	end	duration	product	loc
94	AgentTimeline.At	0.0	[0,75]	[1,76]	[1,76]	--	OBJECT:Home(26)
209	AgentTimeline.Go	1.0	[1,76]	[2,77]	[1,76]	--	--
142	Agent.Buy	2.0	[2,77]	[12,87]	10	OBJECT:Drill(56)	--
308	AgentTimeline.At	2.0	[2,77]	[12,87]	[1,85]	--	OBJECT:HardwareStore(38)
434	AgentTimeline.Go	12.0	[12,87]	[13,88]	[1,76]	--	--
110	Agent.Buy	13.0	[13,88]	[23,98]	10	OBJECT:Milk(50)	--
126	Agent.Buy	13.0	[13,88]	[23,98]	10	OBJECT:Banana(44)	--
240	AgentTimeline.At	13.0	[13,88]	[23,98]	[1,85]	--	OBJECT:SuperMarket(32)
338	AgentTimeline.Go	23.0	[23,98]	[24,99]	[1,76]	--	--
158	AgentTimeline.At	24.0	[24,99]	[25,100]	[1,76]	--	OBJECT:Home(26)

Notice that while EUROPA specifies ranges of times for each event that will satisfy all constraints (for example, buying the drill could start anywhere between timepoint 2 and timepoint 77 inclusive), it has decided that the visit to the hardware store will precede the visit to the supermarket, and has correctly discovered that both milk and bananas can be purchased during the same visit to the supermarket.